

Uncertainty Support in the Spectral Information System SPECCHIO

Andreas Hueni , Member, IEEE, Kimberley Mason, and Simon Trim 

Abstract—The spectral information system SPECCHIO was updated to support the generic handling of uncertainty information in the form of uncertainty tree diagrams. The updates involve changes to the relations database model as well as dedicated methods provided by the SPECCHIO application programming interface. A case study selected from classic field spectroscopy demonstrates the use of the functionality. In conclusion, a database-centric automated uncertainty propagation in combination with measurement protocol standardization will provide a crucial step toward spectroscopy data accompanied by propagated, traceable, uncertainty information.

Index Terms—Information systems, relational databases, spectroscopy, uncertainty.

I. INTRODUCTION

THE guide to the expression of uncertainty in measurement (GUM) states that any measurement without a quantitative indication of quality cannot be compared, neither among themselves nor with a reference standard [1]. While this axiom applies essentially to all measurements, we shall here particularly concentrate on spectroradiometric measurements, which have been classified to be one of the least reliable of all physical measurements [2], [3].

Uncertainty can also be seen as information about measured data, defining its veridicality [4], and it may, therefore, also be classified as a part of the metadata that give context to the measurand and document the process and conditions of measurement [3].

The reporting of uncertainties of spectroradiometer data and derived products in the remote sensing community is however still in its infancy. Airborne and space-based imaging spectrometer missions have started to deal with various sources of uncertainties [5], [6], [7], [8], [9], [10], and have various research groups for in situ measurements [11], [12], [13], [14]. One example are the efforts in the framework of the EUFAR

HYQUAPRO project, which was carried out already a decade ago [7]. However, the community uptake of quality indicators and, in particular, uncertainty propagation and traceability of measurements leaves much to be desired. A recent questionnaire carried out among members of the European COST action SENSECO has demonstrated that the inclusion of uncertainty in the data to information processing flow is still not regularly applied [15].

The reasons for this lag in community uptake are likely manifold. It can, however, be argued that the lack of standardized approaches to the more common data to information transformations and the absence of uncertainty support in remote sensing software packages contribute also significantly to this state of affairs [16]. Recently, the CoMet Toolkit software project has been made available to the metrology community [17]. It is designed to assist practitioners to generically propagate uncertainties and includes support for error covariances. Still, such software requires any user to gain sufficient knowledge in uncertainty analysis and propagation, ideally through remote sensing specific training course material [5].

The situation appears slightly different in the community working on climate data records where users would welcome quality information and highly value traceability chains, but there the problem appears that data providers are not consistent in delivering the required data with an end-to-end metrological traceability [18]. One particular science gap that is of consequence to the present work relates to the lack of availability of in situ measurements with documented metrological traceability [18].

It is, thus, a natural conclusion that spectral information systems should include uncertainty and traceability information and support their handling [14], [19].

The spectral information system SPECCHIO is an advanced software package that was developed by the Remote Sensing Laboratories of the University of Zurich [20], [21], [22], [23] and enhanced in a project funded by the Australian National Data Service [19]. SPECCHIO uses a client-server architecture and a MySQL relational database management system (DBMS) to handle and store point spectrometer data. It supports most stages of the spectroscopy data life cycle through a graphical user interface and an application programming interface (API). At the heart of the system is a flexible and generic metadata concept that is easily enhanced to include new attributes. Rich metadata held by SPECCHIO enables contextual awareness, data sharing, long-term data usage, and queries in metadata space [24].

Manuscript received 10 December 2022; revised 31 January 2023; accepted 16 February 2023. Date of publication 2 March 2023; date of current version 16 March 2023. This was supported in part by the EMPIR Program supported by the Participating States and the European Union's Horizon 2020 Research and Innovation Program under Project 16ENV03 MetEOC-3 and Project 19ENV07 MetEOC-4, and in part by COST Action CA17134 "Optical for spatiotemporal SENSing of Scalable ECophysiological traits" (SENSECO) through its support on networking and dissemination. (Corresponding author: Andreas Hueni.)

The authors are with the Remote Sensing Laboratories, University of Zurich, 8057 Zurich, Switzerland (e-mail: ahueni@geo.uzh.ch; kimberley.mason@geo.uzh.ch; simon.trim@geo.uzh.ch).

Digital Object Identifier 10.1109/JSTARS.2023.3251894

Databases supporting uncertainty in some form have been researched for a while within the realm of computer science [25], [26]. The main research areas involve the modeling of uncertain data, the management of uncertain data, and the mining of uncertain data [27]. All three topics are generally of interest to the application in spectral databases. However, the number of available and operational uncertain DBMSs is small and their implementations are either proprietary or based on a specific relational DBMS implementation [28]. Their use in the context of an already existing information system like SPECCHIO is thus very limited or prohibitive unless a complete redesign were to be planned. Consequently, a solution upgrading the SPECCHIO system to support uncertainties in the context of spectral vector data while maintaining the existing DBMS was required.

In this article, we present the concepts and implementations that enable the SPECCHIO system [19] to store and manage traceability chains and uncertainties. A case study serves to demonstrate the function of the system by applying it to the common use case of the calculation of hemispherical-conical reflectance factors (HCRF) [14], [29] from radiance data.

II. CONCEPTS

The concepts presented in this section are driven by system requirements, which can be summarized as follows.

- 1) Traceabilities can be stored in a generic way in a form that is compliant with uncertainty tree diagrams as introduced by FIDUCEO [30].
- 2) The system offers generic APIs that can define, manipulate, and retrieve the uncertainties and their tree structure.
- 3) The database model stores uncertainty data in an efficient way and avoids redundancy where possible.
- 4) Uncertainty vectors and matrices can be of varying size in the spectral domain, matching the spectral dimension of the measured spectral vectors.

A. Data Model

To introduce the data model, we make use of a simple and yet diverse example of an uncertainty tree diagram (see Fig. 1) to illustrate various cases of uncertainties and their propagation. The chosen example is taken from a straightforward spectro-radiometric calibration and is based on an actual process carried out in the calibration home base (CHB) at the German Aerospace Centre [31]. The goal of this calibration is the provision of radiometric gains (g) and offsets (o) at defined bands for which the spectral response function (SRF) has been defined through spectral calibration. The following description is written with a focus on the calibration and propagation process of a single band for simplicity. In reality, the process applies obviously to all spectral bands with all parameters having individual values per band. The variables introduced as follows are, therefore, vectors with their dimension equivalent to the number of spectral bands of the instrument under calibration.

The spectral calibration (see Spectral Branch in Fig. 1) uses emission line lamps where the wavelengths of the emission

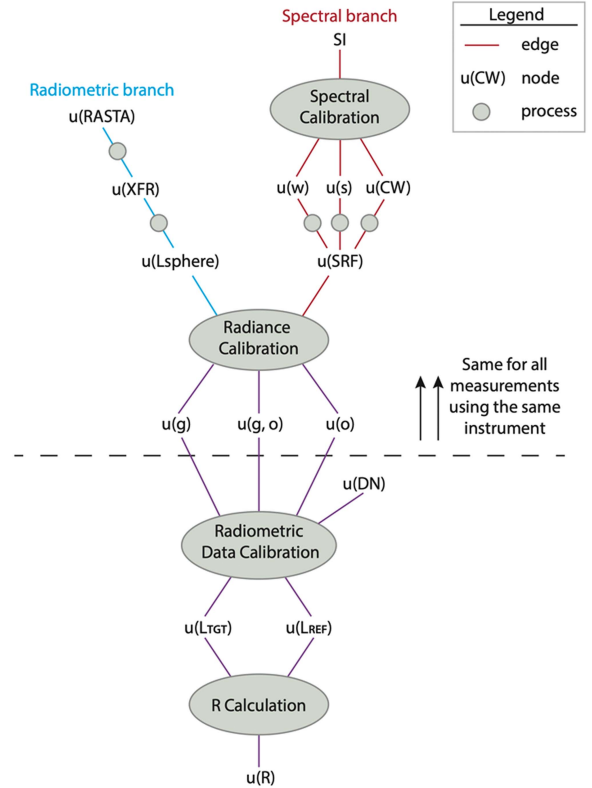


Fig. 1. Simplified example of an uncertainty tree diagram, showing nodes (uncertainties) and edges (propagations through processes) of two branches that combine through the process of radiometric calibration. The boundary between shared instrument calibration uncertainties and individual measurement uncertainties is depicted by a dashed horizontal line.

peaks are given in the Atomic Spectra Database by the National Institute of Standards and Technology (NIST) [32]. The SRF parameters are extracted through a process, which we describe in detail in [33]. The important point is that the uncertainty of the SRF is produced by uncertainty propagation through the spectral calibration algorithm and is, thus, traceable to NIST [32]. The SRF is parameterized by center wavelength (CW), shape (s), and width (w), defining a symmetric super Gaussian sensitivity function, which has been shown to better approximate the true SRF shape for some spectrometers [33]. The SRF has a combined uncertainty $u(SRF)$, computed by propagating $u(CW)$, $u(s)$, and $u(w)$. Such a propagation can be defined analytically according to the GUM [1] as

$$u(SRF)^2 = \left(\frac{\partial SRF}{\partial w} \right)^2 \cdot u(w)^2 + \left(\frac{\partial SRF}{\partial s} \right)^2 \cdot u(s)^2 + \left(\frac{\partial SRF}{\partial CW} \right)^2 \cdot u(CW)^2. \quad (1)$$

The radiometric calibration (see Radiometric Branch in Fig. 1) makes use of an integrating sphere providing a homogenous light field with a radiance L_{Sphere} . L_{Sphere} has been established via a measurement of its intensity by a transfer spectroradiometer (XFR), which in turn has been calibrated against a spectral radiance transfer standard (RASTA) [34]. The radiometric coefficients will, therefore, be traceable to RASTA. For practical

purposes, the depicted traceability chain ends here with an official calibration certificate having a coverage factor $k = 2$, as RASTA itself is traceable to a secondary standard at the Physikalisch-Technische Bundesanstalt. RASTA in turn is traceable to the International System of Units (SI) via a blackbody radiator as German national primary standard [35].

The process of radiometric calibration combines the spectral and radiometric branches. In essence, L_{Sphere} is convolved by the instrument SRF , i.e., L_{Sphere} is resampled to the spectral resolution of the spectroradiometer under calibration, resulting in L_{Sensor} , which is the radiance that the sensor is expected to measure

$$L_{\text{Sensor}} = L_{\text{Sphere}} \cdot SRF. \quad (2)$$

Here, we take the case that different sphere intensities were sampled, allowing to create first order polynomial fits of convolved sphere radiances versus digital numbers (DN) recorded by the instrument under calibration. The radiometric forward model defines the DN as a function of at-sensor radiance L_{Sensor} , integration time Δt , gains g , and offsets o [36]

$$\text{DN} = \Delta t \cdot g \cdot L_{\text{Sensor}} + o. \quad (3)$$

Solving (3) establishes the radiometric calibration coefficients g and o . A propagation of the uncertainties $u(L_{\text{Sphere}})$ and $u(SRF)$ leads to the radiometric coefficient uncertainties $u(g)$ and $u(o)$ and the covariance $u(g, o)$, which describes the correlation of g and o . The correlation coefficient matrix for g and o consists of only four elements, of which the diagonal elements have the value of 1 and the off-diagonal elements are equal. For a propagation of uncertainties according to the law of propagation of uncertainties [1], only one off-diagonal element is required. The final form of $u(g, o)$ is, therefore, also a vector of same dimension as $u(g)$ and $u(o)$. More complex correlation matrices can, however, exist in potentia, which will not reduce to a single value per band.

The next step of uncertainty propagation is applied to the process of radiometric data calibration, i.e., where a at-sensor radiance L is calculated by applying integration time Δt , g , and o to DN obtained from a measurement of a target [36]

$$L_{\text{Sensor}} = \frac{\text{DN} - o}{\Delta t \cdot g}. \quad (4)$$

It is at this point that there occurs a boundary within the uncertainty tree (see the dashed line in Fig. 1). This boundary splits the tree into two sections: 1) an upper part where all uncertainties are due to the calibration and characterization of an instrument and are thus shared by all measurements that are taken with that calibrated instrument, and 2) a lower part where uncertainties are individual per measurement.

The reason for this boundary is the introduction of the measurement noise $u(DN)$ into the radiometric data calibration under the assumption that the noise is a function of at-sensor radiance intensities and, therefore, individual per measurement.

Reflectance factors R get calculated from radiances by normalizing target radiances L_{TGT} with reference panel radiance L_{REF} . Here, we assume an ideal 100% reflective Lambertian reference panel and that no other sources of uncertainties exist

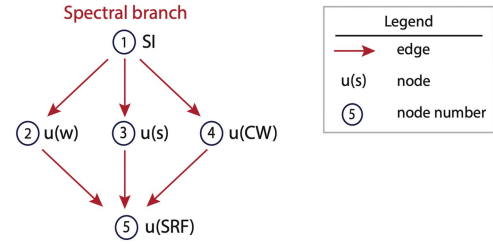


Fig. 2. Subset of an uncertainty tree diagram, showing how the uncertainty of an SRF can be traced back via uncertainties in CW, s , and w parameters to a common SI reference, here the atomic database of NIST [32].

and only $u(L_{\text{TGT}})$ and $u(L_{\text{REF}})$ get propagated to $u(R)$.

$$R = \frac{L_{\text{TGT}}}{L_{\text{REF}}}. \quad (5)$$

The required data model is in essence based on a tree structure built from nodes (uncertainties like $u(SRF)$) and edges (uncertainty propagation processes such as *Radiance Calibration*). The data model must be able to accommodate the following requirements as practically introduced in the above example.

- 1) An uncertainty can either belong to an instrumentation or to a measured or computed spectrum while uncertainty trees can consist of both instrumentation and spectrum level uncertainties.
- 2) An instrumentation can have multiple calibrations, each calibration coming with its own uncertainty tree.
- 3) An uncertainty can be based on several uncertainties that were propagated together, e.g., through the process of radiometric calibration.
- 4) Uncertainties can either be vectors or matrices, the latter in cases where a reduction to diagonal matrices, as introduced by Mittaz et al. [30], is not possible or wished.
- 5) Uncertainties can either be relative or absolute and have an associated coverage factor k .
- 6) A propagation process, i.e., edge, can create several uncertainties, meaning that uncertainties can have a many-to-many relationship.
- 7) Propagation processes can be based on partial derivatives of measurement equations, or they can simply be algorithms that are often too complex to be explicitly derived, in which case Monte Carlo methods are employed [1].
- 8) Spectra can be associated with several uncertainty calculations, e.g., if they are used in different models (e.g., SRF parameterization models), that result in unique uncertainties.

The tree structure is modeled using an adjacency matrix. This concept is introduced here on the example of a subset (see Fig. 2) of the uncertainty tree as shown in Fig. 1. Numbers have been given to the nodes and all propagation processes are shown as simple edges.

The adjacency matrix for this example defines the edges existing between nodes (see Fig. 3). For this purpose, the matrix is always square and its elements are Booleans. Node 1 (NIST) connects to nodes 2 ($u(s)$), 3 ($u(w)$), and 4 ($u(CW)$). Nodes 2, 3, and 4 each connect to node 5 ($u(SRF)$). Node 5 has no further connections.

	1	2	3	4	5
1	0	1	1	1	0
2	0	0	0	0	1
3	0	0	0	0	1
4	0	0	0	0	1
5	0	0	0	0	0

Fig. 3. Adjacency matrix for the tree diagram shown in Fig. 2.

The diagonal elements and the lower triangular matrix remain zero. Directionality of the edges is given by the lookup procedure. Row index lookups traverse down along the direction of uncertainty propagation while column lookups traverse up the tree along the traceability chain, ideally ending at an international standard.

B. Application Programming Interface and Uncertainty Classes

The uncertainty functionality is part of the SPECCHIO API. The SPECCHIO API is implemented in the SPECCHIO client that handles the communication with the SPECCHIO server. End users are, thus, shielded from the database storage details, which are handled by the server. The uncertainty methods of the API take instances of new uncertainty classes as input or deliver them as output.

These uncertainty classes are written generically to handle both vector and matrix data of flexible size, thus supporting any spectral dataset with a specific number of spectral bands.

The Java class `UncertaintySet` represents the broad uncertainty set: a collection of nodes and the links between them in the form of an adjacency matrix. The nodes are represented within the `UncertaintySet` by a “node set ID,” an identifier for a “node set.” Each node set is a collection of nodes pertaining to one adjacency matrix/uncertainty set. It includes information about each node’s node number (i.e., index) within the rows/columns of the adjacency matrix.

An individual uncertainty node can be represented by the class `UncertaintyNode`. This class includes information about the uncertainty vector, confidence level, and, for absolute uncertainty values, the associated unit type of a given node. Two classes, `UncertaintySpectrumNode` and `UncertaintyInstrumentNode`, inherit `UncertaintyNode` and contain specific characteristics of spectrum level and instrument level nodes, respectively.

The SPECCHIO API has been extended to allow users to create new uncertainty sets and then subsequently add details of uncertainty nodes to the sets with associated edge relationships. New functions have also been added to also retrieve uncertainty information.

As mentioned above, a spectrum might be used in multiple uncertainty sets. These relationships are defined by registering a spectrum in an `UncertaintySpectrumNode` object. To facilitate the generation of unique matrices holding uncertainty information for spectral collections, we have upgraded the SPECCHIO Space Factory [19]. The Space Factory generates congruent

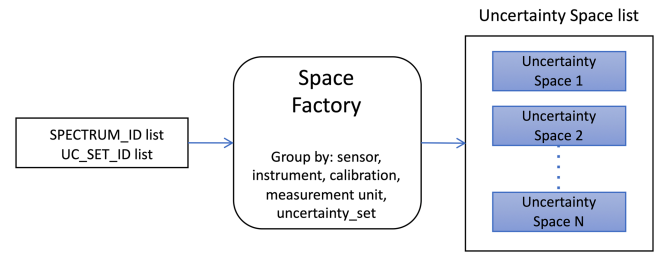


Fig. 4. Building of uncertainty spaces via the SPECCHIO space factory.

spectral spaces, i.e., each space contains spectra that have identical measurement units and were captured by a specific instrument with a defined calibration. In the case of uncertainty spaces, the additional criterion of a matching uncertainty set is added (see Fig. 4) and the result is a list of uncertainty spaces containing references to their respective uncertainty sets and spectra. The order of uncertainty vectors within the space is based on a metaparameter-based sorting approach. Applying the same sorting setting for both spectral and uncertainty space generations guarantees that the vectors in the resulting matrices are aligned.

C. Database Model

New tables were added to the existing SPECCHIO MySQL database in order to store uncertainty information. The schema of tables (see Fig. 5) shows the new tables, including primary and foreign key links. The schema has been designed in such a way as to maintain links between uncertainty information while reducing redundancy.

The `uncertainty_set` table is the starting point for building a new uncertainty tree. It stores the adjacency matrix and the ID of the associated collection of nodes in `uncertainty_node_set`. The adjacency matrix is of type “Matrix” within Java basing on the UJMP package [37] and is stored in MySQL as a binary large object (BLOB).

We decided to extend the Boolean model for adjacency matrix values in order to store more detail about the uncertainty propagation process. While “0” indicates no link and “1” indicates a “simple link” between two nodes, numbers greater than 1 are possible and are stored in the `uncertainty_edge` table. In order to describe a propagation, an “edge_value,” which contains a description of the edge process, is assigned to the next available `edge_id` integer. The link between `edge_id` and `uncertainty_set` is an implicit link.

The design choice to store the adjacency matrix as BLOB was governed by the goal to simplify the storage of the matrix and make it easily available within the Java layers of the SPECCHIO system. A simple deserialization process, thus, yields an object of type UJMP Matrix. The downside of this approach is that a direct SQL query of the adjacency matrix is not possible. Consequently, maintenance of the adjacency matrix data integrity cannot be implemented via SQL triggers but must be maintained by the logic implemented in the SPECCHIO server code. This follows the philosophy of the SPECCHIO system that abstracts

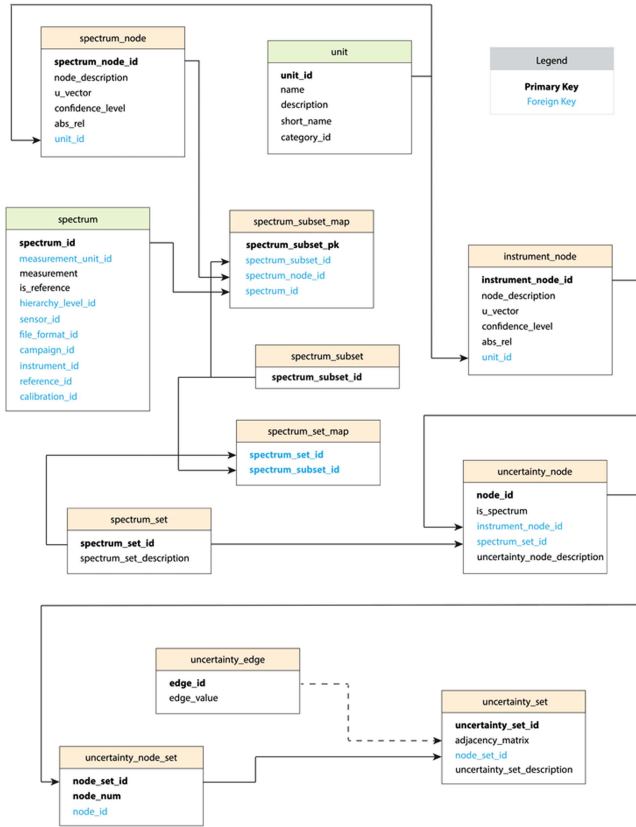


Fig. 5. Database model. The tables with green headers are tables that already existed in SPECCHIO. Those with orange headers are new tables related to uncertainty. The overall SPECCHIO schema is omitted for clarity.

the database storage via the server-based code, allowing end user access via APIs.

The `uncertainty_node_set` table contains a collection of nodes and their coordinates within the adjacency matrix rows and columns (`node_num`).

A given node in the uncertainty tree might be associated with many spectra, each with its own entry in the `spectrum_node` table. To avoid overcomplicating the uncertainty tree, we group these spectrum nodes into a “`spectrum_set`” and then assign this set to a single entry in the “`uncertainty_node`” table. An entry in the “`uncertainty_node`” table could instead relate to an instrument node (a 1:1 relationship). In this case, the “`is_spectrum`” flag would be set to 0.

The `spectrum_subset` table allows us to provide nuance to a spectrum set by splitting it into subsets. This is useful in cases where we have spectra split into target and reference measurements and wish to store them as such.

III. CASE STUDY

The case study applies the newly developed uncertainty support of SPECCHIO to the calculation of HCRF from radiance measurements.

To get the reader familiar with the context, we first introduce the measurement protocol followed by a description of the processing approach. This is the basis for the development of the

uncertainty tree diagram, where we introduce various sources of uncertainty. We then give examples of how nodes and edges can be entered into SPECCHIO using the API. Finally, we show how the uncertainty tree can be interrogated via the API to produce a spectral plot showing all sources of uncertainty identified in the tree diagram that contribute to an HCRF estimate.

A. Abbreviations

TABLE I
LIST OF SPECIFIC ABBREVIATIONS USED IN SECTION III

Abbreviation	Term
ASD	Analytical Spectral Devices company, also used to designate the instruments built by this company.
NIR	Near-Infrared
QE	Quantum Efficiency
REF	White Reference
SGCP	Spectral Ground Control Point
SWIR	Short-wave infrared
SWIR1	First SWIR detector of the ASD, covering 1000–1800 nm
SWIR2	Second SWIR detector of the ASD, covering 1801–2500 nm
TGT	Target
UV	Ultraviolet
VNIR	Visible Near-Infrared. Also used to designate the first detector of the ASD, covering 350–999 nm

B. Measurement Protocol

The measurement protocol introduced here has been specifically developed at the remote sensing laboratories to be used during field campaigns for vicarious calibration and validation (CAL/VAL) of airborne imaging spectroscopy data [14]. We refer to this type of data as Spectral Ground Control Points (SGCP) [19]. Measurements with analytical spectral device (ASD) spectroradiometers are carried out in radiance mode, as endorsed by Milton et al. [3], and comprise the following steps.

- 1) A Spectralon white reference (REF) panel mounted on a tripod is leveled using a bubble level.
- 2) The spectroradiometer is optimized for integration time and internal gains while pointing at the REF panel, thus achieving the best signal-to-noise ratio for the current irradiance conditions.
- 3) Five REF measurements are taken.
- 4) 30 target (TGT) measurements are taken while moving the optic over the TGT to capture its variability.
- 5) Five REF measurements are taken.

C. Database-Centric Radiance to HCRF Processing

The first step toward HCRF data is to ingest the radiance files into the SPECCHIO. Details on the loading process and the metadata augmentation of SGCP data with an average of 36 metaparameters per spectrum are given by Hueni et al. [19].

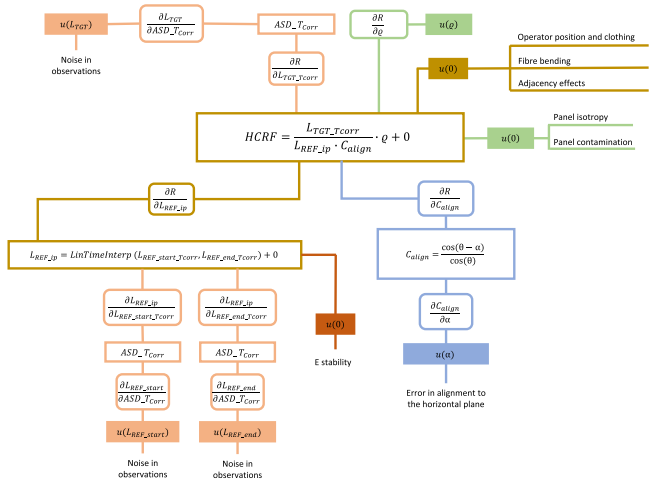


Fig. 6. Uncertainty tree diagram of HCRF data, showing contributions of temperature corrections, linear temporal interpolation of reference panel readings, reference panel reflectance uncertainty, and uncertainty due to panel alignment.

These radiances data are then further processed in a Matlab-based tool that interfaces with the SPECCHIO system using the SPECCHIO API. The processing steps are as follows.

- 1) Automated identification and flagging of REF and TGT spectra based on their spectral content
- 2) Correction of interchannel, i.e., interdetector, radiometric steps based on an instrument thermal model and storage of corrected radiance spectra in SPECCHIO [38]. The following steps use the corrected radiance spectra.
- 3) Estimation of REF radiances for all TGT radiances by a linear interpolation over time between the bracketing REF radiance sets of five spectra each.
- 4) Computation of absolute HCRF

$$\text{HCRF} = \frac{L_{\text{TGT}}(t)}{L_{\text{REF_ip}}(t)} \cdot \rho \quad (6)$$

where $L_{\text{TGT}}(t)$ is a TGT radiance measured at time t , $L_{\text{REF_ip}}(t)$ is the REF radiance interpolated over time at the instant t , and ρ is the reflectance factor of the REF panel. Also note that ρ is given as a reflectance and thus independent of illumination and observation geometries. This is a simplification, based on the assumed ideal isotropic property of the Spectralon panel [39]. In reality, the bidirectional reflectance distribution function of the panel should be taken into consideration [39].

- 5) Storage of the calculated HCRF spectra in SPECCHIO.

D. Uncertainty Tree Diagram

The uncertainty tree diagram shown in Fig. 6 is grouped around the primary measurement equation that calculates the reflectance factor R , which is an HCRF for the field case under natural illumination conditions [29]. Starting from the terms in this equation, we will follow each branch of uncertainty. In these diagrams, the edges representing uncertainty propagations are written explicitly as partial derivatives, thus defining the sensitivity coefficients [12], [30].

Starting with the $L_{\text{TGT_Tcorr}}$ term, it can be seen that R is sensitive to an uncertainty of ASD_Tcorr , which stems from propagating the uncertainty $u(L_{\text{TGT}})$ through the ASD Temperature Correction (ASD_Tcorr) [38]. The uncertainty $u(L_{\text{TGT}})$ is defined by the measurement noise.

One may notice at this point that the uncertainties due to radiometric gain and offset, as introduced in the data model example earlier, are not listed as sources of uncertainty for $u(L_{\text{TGT}})$. They are here omitted by assuming radiometric linearity of the system with the offset term being zero [3]. This is in fact often applied to field spectroradiometers where only a gain factor is given. The terms $L_{\text{TGT_Tcorr}}$ and $L_{\text{REF_ip}}$ could then be written explicitly as $DN_{\text{TGT_Tcorr}} \cdot g$ and $DN_{\text{REF_ip}} \cdot g$. As the gains do cancel out [see (6)], so does the uncertainty, which is related to the radiometric calibration.

The term $L_{\text{REF_ip}}$ has an uncertainty stemming from the linear interpolation over time, *LineTimeInterp*. This linear interpolation is calculated from REF measurements taken before ($L_{\text{REF_start_Tcorr}}$) and after ($L_{\text{REF_end_Tcorr}}$) the TGT measurements. $L_{\text{REF_start_Tcorr}}$ and $L_{\text{REF_end_Tcorr}}$ have both corrected for temperature-induced radiometric interchannel steps by ASD_Tcorr , which are again connected to the uncertainties related to the noise of the measurements $L_{\text{REF_start}}$ and $L_{\text{REF_end}}$.

The term $L_{\text{REF_ip}}$ is also influenced by the stability of the irradiance as the assumption is that the irradiance E changes linearly over time between the two REF sequences. This linearity is a fair approximation, as a full measurement sequence can typically be completed in about 3 min by a skilled operator, and the sun zenith will only change by N arc minutes during that time, with N being a function of latitude, local hour angle, and solar declination [40]. The uncertainty of E is, therefore, mainly caused by perturbations of the irradiance field during these few minutes by atmospheric effects, such as subvisual clouds passing in front of the solar disc [3]. The uncertainty is, thus, specified as $u(0)$, as we have no readily available model at this point to estimate the short term uncertainty of E .

The REF panel reflectance ρ has an associated uncertainty $u(\rho)$, which is provided by the manufacturer of the Spectralon panels as a part of the calibration certificate with a coverage factor of $k = 2$.

The final term of the primary equation is c_{align} . It is linked to the errors in horizontal alignment of the REF panel, where horizontal means that the surface of the panel should be perpendicular to the direction of gravity [41]. The angular alignment uncertainty $u(\alpha)$ is propagated to the equation for the c_{align} , parameterized by the solar zenith angle. It is obvious that for a zero alignment error, c_{align} would assume a value of 1. We are using here a single alignment error with a tilt direction in the solar principal plane [14] but this could further be refined by introducing two tilt angles [42] or an azimuth and a tilt angle.

The $+ 0$ in the primary equation is used to list other sources of uncertainties that are not yet explicitly addressed in the uncertainty analysis. This includes the simplifying assumption that the panel is isotropic in its reflective behavior and not contaminated or degraded and that the operator position and clothing, bending of the fiber, and further adjacency effects do

not affect R . These assumptions are obviously wrong but have been selected to constrain the complexity.

E. Definitions of Nodes and Edges in SPECCHIO

Code was written in Matlab to define uncertainty vectors, propagate these uncertainties for an SGCP dataset, and insert the nodes and edges into SPECCHIO using newly developed API functions. The practical use of these new classes and methods is shown in the following sections.

1) Creation of a New Uncertainty Set

The creation of a new uncertainty set in the database is the first step before the set can be defined by adding nodes and defining edges. The set description can be any alphanumeric string.

```
uc_set = UncertaintySet();
uc_set.uncertainty_set_description = 'uR_example';
uc_set_id = specchio_client.insertNewUncertaintySet(uc_set);
```

2) $u(L)$ Due to Noise

The noise of all ASD radiance measurements, i.e., encompassing L_{TGT} , L_{REF_start} , and L_{REF_end} , was estimated by using an empirical noise model that was developed based on laboratory characterizations. It describes the measurement noise as a linear function of at-sensor radiance per spectral band. The effect of the ASD internal average setting on noise is modeled by assuming a reduction of noise with \sqrt{N} , where N is the number of internal averages.

$u(L)$ assumes the form of a matrix due to the dependence on at-sensor radiance intensity of the noise model, thus containing one specific uncertainty vector per radiance spectrum. $u(L)$ is entered using the code shown as follows. An `UncertaintySpectrumNode` object is created and parameterized to refer to the database identifiers of the radiance spectra given in the Java ArrayList L_ids . The $u(L)$ matrix is entered as uncertainty vectors of the node and the `uncertainty_type` property is set to relative. The uncertainty source is a text description, referring to the instrument noise model described above. This set is then added to the database via the `insert_uncertainty_of_set` client method. It must be noted that the order of the uncertainty vectors in the matrix matches the order of the `spectrum_ids` to keep the uncertainty data aligned with the spectral information.

```
spectrum_node = UncertaintySpectrumNode();
spectrum_node.spectrum_ids = L_ids;
spectrum_node.abs_rel = 'rel';
spectrum_node.confidence_level = 1;
spectrum_node.uncertainty_node_description = 'u(L) due to noise';

spectrum_node.setUncertaintyMatrix(uL);

uL_noise_uncertainty_node_id =
specchio_client.insertUncertaintyNode(spectrum_node, uc_set_id);
```

3) $u(ASD_Tcorr)$

A simple uncertainty estimate of the ASD_Tcorr was established during this work. Only the noise of the data to be corrected by the model output was included, as a complete uncertainty analysis of the temperature experiment and its propagation to the

temperature correction model would warrant its own dedicated study and publication.

In essence, the correction algorithm assumes that the SWIR1 detector is quasi-stable and that a state of the thermal model can be found which minimizes the radiometric discontinuity at the borders of the neighbouring channels, i.e., VNIR and SWIR2 [38]. Any noise in the data impacts this matching approach and an error in thermal model choice due to noise at the channel splicing position is propagated to the full channel. This propagation of noise to the corrected radiance spectra was implemented as a Monte Carlo analysis, based on the noise model of ASD radiance data introduced above.

The result is a matrix holding the uncertainty vectors per spectrum. These uncertainties are combining the measurement noise and the uncertainty of the temperature correction due to the implicit propagation of the uncertainty by the Monte Carlo approach.

At this point, we decide that we would like to split the traceability tree into two branches, treating the TGT and REF uncertainties separately. We choose to do so because it better fits our uncertainty tree diagram, as shown in Fig. 6.

The insert into the database links subsets of this matrix to the temperature corrected radiance spectra, specified by $L_{TGT_ids_T_corr}$ and $L_{REF_ids_T_corr}$. Two uncertainty sources are added here. One is the process that propagates uncertainties and the second is the input uncertainties of the process, i.e., the spectral set of the radiance measurement noise $u(L)$ that got inserted in the previous section. The two new uncertainty sets, thus, refer to an uncertainty source that contains the noise for both REF and TGT radiance spectra.

```
% Creation of two subsets for TGT/REF
spectrum_node = UncertaintySpectrumNode();
spectrum_node.abs_rel = 'rel';
spectrum_node.confidence_level = 1;
spectrum_node.node_description = 'ASD T Correction TGT';
spectrum_node.spectrum_ids = L_TGT_ids_T_corr;
spectrum_node.setUncertaintyMatrix(uT_corr_rel_TGT);

uT_corr_tgt_subset_id =
specchio_client.insertSpectrumSubset(spectrum_node);

spectrum_node = UncertaintySpectrumNode();
spectrum_node.abs_rel = 'rel';
spectrum_node.confidence_level = 1;
spectrum_node.node_description = 'ASD T Correction REF';
spectrum_node.spectrum_ids = L_REF_ids_T_corr;
spectrum_node.setUncertaintyMatrix(uT_corr_rel_REF);

uT_corr_ref_subset_id =
specchio_client.insertSpectrumSubset(spectrum_node);

% Creating an arraylist of subset ids
uT_corr_subset_id_list =
get_INT_jArrayList([uT_corr_tgt_subset_id,
uT_corr_ref_subset_id]);

% Inserting uncertainty node using subsets
spectrum_node = UncertaintySpectrumNode();
spectrum_node.spectrum_subset_ids = uT_corr_subset_id_list;
spectrum_node.uncertainty_node_description = 'u(L Tcorr)';
spectrum_node.add_uncertainty_source_by_id(uL_noise_uncertainty_node_id, 'ASD T Correction'); % an edge connection

uT_corr_uncertainty_node_id =
specchio_client.insertUncertaintyNode(spectrum_node, uc_set_id);
```

4) $u(LinTInterp)$

The uncertainty of the linear temporal interpolation of the REF radiances was established by a Monte Carlo analysis,

propagating the uncertainty of the temperature corrected radiances. The result is a matrix with uncertainties of the interpolated REF radiances, coinciding temporally with the TGT radiances, i.e., they are uncertainties of the estimates of the REF radiance at the time of the capture of the TGT spectra.

The matrix `uLref_interp_rel` is inserted into the database as a new `UncertaintySpectrumNode`, not specifying any spectrum identifiers because the interpolated REF radiances are computed on the fly and not inserted into the database. We account for the interpolation of REF radiances and related uncertainties in the tree diagram, which is visualized in Fig. 6. The sources of uncertainty are specified by the uncertainty set of the temperature-corrected REF radiances and as an algorithm of the linear temporal interpolation.

```
spectrum_node = UncertaintySpectrumNode();
spectrum_node.abs_rel = 'rel';
spectrum_node.confidence_level = 1;
spectrum_node.node_description = 'u(Lref_interp)';
spectrum_node.add_uncertainty_source_by_id(uT_corr_uncertainty_node_id, 'Irrad. linear time interp.');
```

```
edge_description
spectrum_node.setUncertaintyMatrix(uLref_interp_rel);

uL_REF_interp_uncertainty_node_id =
specchio_client.insertUncertaintyNode(spectrum_node, uc_set_id);
```

5) $u(\varrho)$

The reflectance ϱ of the Spectralon panel and its uncertainty $u(\varrho)$ were loaded from a calibration file supplied by Labsphere at the time of purchase. The uncertainty of the panel is then added as an object of type `UncertaintyInstrumentNode`.

```
instrumentation_node = UncertaintyInstrumentNode();
instrumentation_node.abs_rel = 'abs';
spectrum_node.confidence_level = 1;
instrumentation_node.node_description = 'u(Spectralon)';
instrumentation_node.unit_id =
ch.specchio.spaces.MeasurementUnit.Reflectance;
instrumentation_node.setUncertaintyVector(u_spectralon);
```

```
uSpectralon_id =
specchio_client.insertUncertaintyNode(instrumentation_node,
uc_set_id);
```

Note that instrumentation calibrations in SPECCHIO can have associated uncertainty vectors, which can be entered by using the SPECCHIO Instrumentation Metadata Editor or loaded by using the SPECCHIO API. At the time of writing, these have not yet been connected with the new generic version for storing uncertainties.

6) $u(c_{align})$

The uncertainty due to the alignment of the reference panel was calculated by using a sun zenith angle of 46.77° and an angular uncertainty of 1° . The latter value is our current estimate of angular errors when carefully aligning a panel using a bubble level [14]. The resulting uncertainty of the alignment factor c_{align} is added to the database by first converting the scalar `uC_align_rel` to a vector of a length equal to the number of bands of the ASD spectrometer and then creating a `UncertaintySpectrumNode` object where the uncertainty source is the process of angular Spectralon alignment. No spectrum identifiers are specified as this uncertainty does not apply to

particular measurements but is only added during the HCRF computation.

```
uL_panel_vector = ones(size(spectra.wvl)) * uC_align_rel;
spectrum_node = UncertaintySpectrumNode();
spectrum_node.abs_rel = 'rel';
spectrum_node.confidence_level = 1;
spectrum_node.node_description = 'u(Angular Spectralon alignment)';
spectrum_node.uncertainty_node_description = 'u(Angular Spectralon alignment)';
spectrum_node.setUncertaintyVector(uL_panel_vector);

uL_panel_alignment_uncertainty_node_id =
specchio_client.insertUncertaintyNode(spectrum_node, uc_set_id);
```

7) $u(R)$

The uncertainty of the reflectance factor R is calculated by first calculating the sensitivity coefficients via the partial derivatives of the measurement equation and then adding the uncertainty contributions in quadrature before taking the square root according to the GUM [1]

$$u(R) = \left(\left(\frac{\varrho}{L_{REF_{ip}} \cdot C_{align}} \right)^2 \cdot u(L_{TGT_T_{corr}})^2 + \left(-\frac{L_{TGT_T_{corr}} \cdot \varrho}{L_{REF_{ip}}^2 \cdot C_{align}} \right)^2 \cdot u(L_{REF_{ip}})^2 + \left(\frac{L_{TGT_T_{corr}}}{L_{REF_{ip}} \cdot C_{align}} \right)^2 \cdot u(\varrho)^2 + \left(-\frac{L_{TGT_T_{corr}} \cdot \varrho}{L_{REF_{ip}} \cdot C_{align}^2} \right)^2 \cdot u(C_{align})^2 \right)^{1/2}. \quad (7)$$

The final propagated uncertainty is a matrix with an individual uncertainty per R vector. The `UncertaintySpectrumNode` instance references the spectrum identifiers of the R spectra and links to the following sources of uncertainty already defined in the database:

- 1) the panel reflectance uncertainty identifier;
- 2) the REF temporal linear interpolation set;
- 3) the set of propagated uncertainty of the temperature correction of the targets;
- 4) the set describing the uncertainty due to panel alignment.

```
spectrum_node = UncertaintySpectrumNode();
spectrum_node.spectrum_ids = R_ids;
spectrum_node.abs_rel = 'rel';
spectrum_node.confidence_level = 1;
spectrum_node.node_description = 'uR';
spectrum_node.uncertainty_node_description = 'uR';
spectrum_node.setUncertaintyMatrix(uR_rel); % relative uncertainty
spectrum_node.add_uncertainty_source_by_id(uT_corr_uncertainty_node_id, 'R estimation');
spectrum_node.add_uncertainty_source_by_id(uL_REF_interp_uncertainty_node_id, 'R estimation');
spectrum_node.add_uncertainty_source_by_id(uL_panel_alignment_uncertainty_node_id, 'R estimation');
spectrum_node.add_uncertainty_source_by_id(uSpectralon_id, 'R estimation');
```

```
uR_uncertainty_node_id =
specchio_client.insertUncertaintyNode(spectrum_node, uc_set_id);
```

8) Data Insert Speed

The time required to define and insert the nodes and edges as defined above was tested in a loop with 100 iterations to

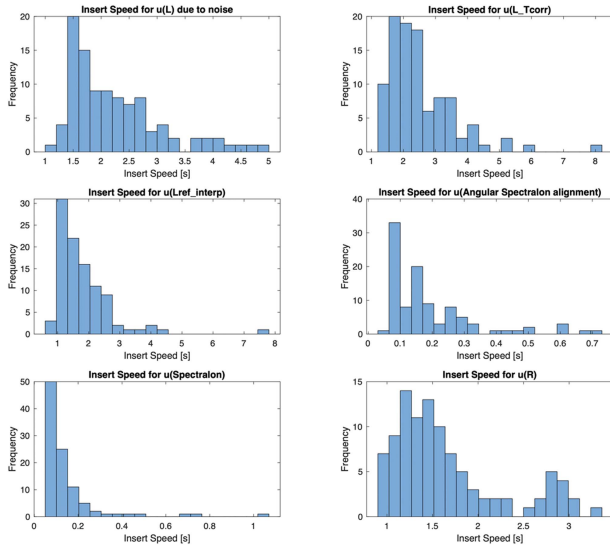


Fig. 7. Histograms of insert times for the six uncertainty nodes of the case study.

gather mean and standard deviation data. Times were measured in Matlab using the tic and toc functions. The insert process was run on a MacBook Pro with a 2.7 GHz Quad-Core Intel Core i7 processor and 16 GB of RAM. The network speed was measured to be 32 Mb/s for downloads and 27.7 Mb/s for uploads. Connection to the SPECCHIO server running on the University of Zurich Science Cluster was established via virtual private network. Average insert times for the uncertainty nodes range around 2 s for nodes with 30 vectors, while nodes with single vectors, such as the Spectralon panel uncertainty $u(\varrho)$, take about 0.15 s (see Fig. 7).

9) Uncertainty Components and Propagated Uncertainty of HCRF

A side benefit of a rigorous uncertainty analysis is the knowledge about the contributions of the sources of uncertainty to a propagated, i.e., combined uncertainty. For spectral applications, such information is easily visualized in a spectral plot, as shown in Fig. 8. For this figure, we plotted the uncertainty components already weighted by their related sensitivity coefficients to allow a better understanding of the impact of the sources of uncertainty as terms of (7) on the combined uncertainty, i.e., the plotted contributing uncertainty vectors were computed as

$$u_{\text{weighted}} = \sqrt{c^2 \cdot u^2} \quad (8)$$

where c is the sensitivity coefficient.

The y-axis of Fig. 8 is set to a logarithmic scale to allow discerning minute details in the SNR related features. In the following paragraphs, we describe all components, providing details about their spectral shapes and the underlying reasons.

The uncertainty vector $u(L_TGT_Tcorr)$ shows generally low uncertainties with an average of 0.6% but contains many features that can be linked to either instrument effects or atmospheric absorptions. Higher uncertainties are found in the UV region, at the end of the VNIR and SWIR2 detectors and at water vapor absorptions near 1140, 1380, and 1900 nm. Other peaks

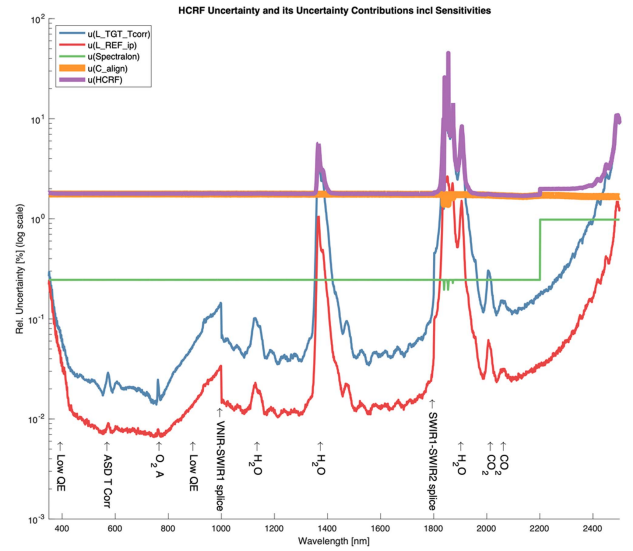


Fig. 8. Mean propagated uncertainty of HCRF and the mean contributing uncertainty vectors per uncertainty source including their sensitivities to HCRF, visually confirming that the angular reference panel alignment is the largest source of uncertainty within the HCRF uncertainty budget. Spikes in the uncertainty vectors are caused by regions of the measured electromagnetic spectrum containing low energy, such as in water vapor absorptions bands around 1350 and 1900 nm, or by instrument related effects such as QE. The coverage factor k of the uncertainty vectors is set to 1.

in the uncertainty spectrum appear at 761 nm caused by oxygen absorption (O_2 A band) and near 2005 and 2062 nm, which can be linked to CO_2 absorptions. These higher uncertainties are resulting from a lower SNR, which is driven by low irradiance intensities reaching the Earth surface at these wavelengths, in combination with low detector quantum efficiencies (QE) for some wavelengths regions, typically encountered for the silicon photodiode based VNIR channel in the UV below 400 nm and NIR above 850 nm. The bowl shape of the uncertainty of the VNIR channel is in essence an inverted QE curve where high QE is found in the middle of the linear array detector detecting wavelength between 500 and 800 nm, gradually decreasing toward the edges. A step in the uncertainty vector at 1000 nm indicates the splicing position of VNIR and SWIR1 detectors of the ASD instrument, while a second step at 1800 nm indicates the joint between SWIR1 and SWIR2 detectors. A further peak near 572 nm can be traced to a higher uncertainty in the ASD temperature correction model.

The uncertainty of the interpolated white reference spectra $u(L_REF_ip)$ follows a similar pattern as $u(L_TGT_Tcorr)$ but has lower uncertainties. This is due to the better SNR provided by the signal of a solar irradiance reflected by a Spectralon panel, and, in particular, due to the linear interpolation. The application of the interpolation effectively reduces the uncertainty. This reduction is the result of the least-squares-based curve fitting procedure where the noise of each fitted measurement point has a limited impact on the resulting fit. This effect is similar to that of an averaging function which also reduces the uncertainty compared to a single measurement.

The uncertainty of the Spectralon panel ($u(Spectralon)$) shows a step function at 2200 nm. This is caused by the uncertainty of the detectors used to calibrate the panel. The detector used for

wavelengths above 2200 nm exhibits a higher uncertainty, which propagates to the values reported in the calibration certificate of the panel.

The panel alignment uncertainty $u(C_{\text{align}})$ is in essence a level line, only slightly modified by its sensitivity coefficient and contributing significantly to the overall uncertainty.

The combined uncertainty of the reflectance factor $u(\text{HCRF})$ is the uncertainty spectrum with the highest amplitude within this plot, which must be the case for a propagated uncertainty. The spectral shape is governed by the contributing uncertainty sources. Its main features are the higher uncertainties in the water vapor bands and at the end of the SWIR2, which are inherited from the measured TGT and REF spectra. A small increase in uncertainty at 2200 nm is contributed by the Spectralon panel. The overall amplitude of $u(\text{HCRF})$ is driven by the panel alignment uncertainty.

F. Retrieving Uncertainties and Displaying the Overall Uncertainty Budget

1) *Retrieving the Uncertainty Sets of a Collection of Spectra:* The uncertainty sets of a collection of spectra can be retrieved via the `getUncertaintySetSpectraLists` API function. As a spectrum can be included in multiple uncertainty sets, e.g., if it is used in different computations leading to unique uncertainty trees, this function groups the spectra by the uncertainty sets. As a result, it returns a list of objects that contain the `uncertainty_set_id` and a list of `spectrum_ids` that belong to the respective uncertainty set.

```
spectrum_id_list = get_INT_jArrayList([35419, 35420]);
uc_lists=specchio_client.getUncertaintySetSpectraLists(spectrum_id_list);
uc_set_list = uc_lists.get(0); % select first set
```

2) *Retrieving Uncertainty Spaces of a Collection of Spectra:* We use the `getUncertaintySpaces` API function to retrieve uncertainty vectors for given spectrum IDs and uncertainty set IDs. This function returns a list of spaces, one space for each uncertainty set. Each space contains a matrix of uncertainty vectors for the spectrum IDs in question as well as the associated uncertainty set ID. Uncertainty vectors can then be used to construct plots that provide the uncertainty of the sources and the total propagated uncertainty.

```
uc_set_id_list =
get_INT_jArrayList(uc_set_list.uncertainty_set_id);
uc_spaces =
specchio_client.getUncertaintySpaces(spectrum_id_list,
uc_set_id_list, 'Acquisition Time');

% get first, end in this case only space
uc_space = specchio_client.loadSpace(uc_spaces(1));

figure
plot(uc_space.getAverageWavelengths,
uc_space.getVectorsAsArray);
title('HCRF Relative Uncertainties');
```

3) *Retrieving Uncertainty Information for a Given Uncertainty Set:* We can retrieve all information stored in an uncertainty set and use this to plot an uncertainty tree. The tree allows a user to see visually each of the nodes and edges. The `getUncertaintySet` function takes a given uncertainty set ID and returns the adjacency matrix, the node numbers, i.e.,

the index of the nodes in the adjacency matrix, and descriptions. For numbers greater than 1 in the adjacency matrix, we use `getEdgeValue` to retrieve their description. Information about nodes can be retrieved via the `getUncertaintyNodeSubSets` function. In the example code as follows, we use this to identify nodes that comprise subsets, such as the REF and TGT subsets of the temperature corrected radiances, and indicate the existence of subsets by modifying the node label string.

The digraph MATLAB function takes an adjacency matrix and a set of node names to plot the node–edge relationships (see Fig. 9).

```
uc_set =
specchio_client.getUncertaintySet(uc_space.uncertainty_set_id);

% Get adjacency matrix and node information
adj_m = double(uc_set.getAdjacencyMatrixAsIntArray());
node_nums = uc_set.getNodeNums();
node_names = uc_set.getUncertaintyNodeDescriptions();

% Convert to cell and indicate subsets
for i=1:size(node_names)
    node_num = node_names.indexOf(node_names.get(i-1));
    node_id = uc_set.getUncertaintyNodeIds.get(node_num);
    ucns = specchio_client.getUncertaintyNodeSubSets(node_id);

    node_names_cell{i} = escape_(node_names.get(i-1));
    if length(ucns) > 1 % Add subset indicator
        node_names_cell{i} = ...
            [node_names_cell{i} ' [' num2str(length(ucns)) ']'];
    end
end

% Extracting values > 0 from adjacency matrix
% When > 1, will have unique edge value in uncertainty_edge
table
adj_m_details.edge_ids = adj_m(adj_m > 0);

for i = 1:length(node_nums)
    edge_id = adj_m_details.edge_ids(i);
    if edge_id == 1
        edge_value = ' ';
    else
        edge_value = specchio_client.getEdgeValue(edge_id);
    end
    adj_m_details.edge_values(i) = string(edge_value);
end

edge_values_cell = cellstr(adj_m_details.edge_values);

% Change all values >1 --> 1
adj_m(adj_m>1) = 1;

% Construct digraph.
figure;
G = digraph(adj_m, node_names_cell);

% Assign names to the edges
G.Edges.name = escape_(edge_values_cell');
h = plot(G, 'EdgeLabel',G.Edges.name);
```

4) *Retrieving Uncertainty Nodes:* Uncertainty nodes can also be retrieved via their node IDs available from the uncertainty set. This allows accessing nodes that are not linked to spectral data via spectrum identifiers and hence cannot be retrieved via the `getUncertaintySpaces` API function. One example is the Spectralon uncertainty node. The code as follows gets the node number from a search of the node names and uses that index to get the uncertainty node identifier via the `getUncertaintyNodeIds` method of the uncertainty set. This ID can then be used to retrieve a list of nodes that belong to that uncertainty node via `getUncertaintyNodeComponents`. A list is returned, as an uncertainty node can have a one-to-many relationship to individual spectrum uncertainties. In the case of

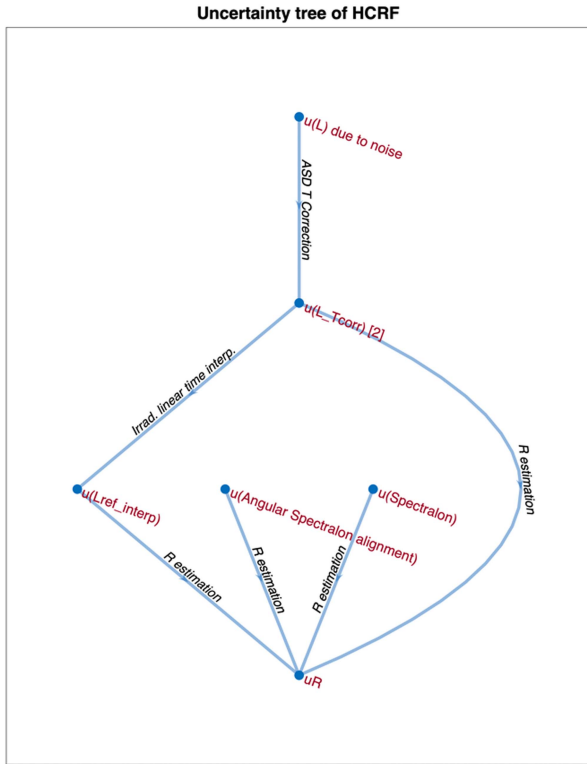


Fig. 9. Automatically constructed uncertainty tree diagram based on uncertainty data stored in SPECCHIO. The displayed node and edge labels represent the string values defined during uncertainty data inserts. Note that the $u(L_Tcorr)$ node comprises two spectral subsets, indicated by the number two in brackets.

the Spectralon node, a single entry is returned as it is an instrument node. The node object contains the uncertainty vector, the confidence interval, and the node description, essentially holding all information given during the node definition prior to inserting it into the database.

```
node_num = node_names.indexOf('u(Spectralon)');
node_id = uc_set.getUncertaintyNodeIds.get(node_num);
ucs = specchio_client.getUncertaintyNodeComponents(node_id);
spectralon_node = ucs(1);
```

IV. DISCUSSION

The presented SPECCHIO system update represents, to our knowledge, the first implementation of uncertainty support in a spectral information system within the remote sensing sciences. Further improvements will be required to increase its usefulness and to keep up to date with new developments and requirements alike, stemming from active research in metrology and related activities in Earth observation.

Future upgrades will have to focus on the inclusion of correlation between spectral bands in order to define the degree of random and systematic uncertainties. This information will be of importance in aiding the propagation of uncertainties provided by the system. Furthermore, the storage of covariances between variables must be considered to fully describe analytical uncertainty propagations. This also points to the potential requirement of storing the probability distribution function (PDF) as Gaussian PDFs are not always reflecting the true distribution.

Finally, sensitivity coefficients or code to propagate uncertainties may also warrant an inclusion in the data model.

The existence of uncertain DBMSs was alluded to in Section I. A review of the existing systems led to our decision to implement our own uncertainty support. The reasons for this are manifold, including the problem of adapting an existing schema, data, and query code to a new DBMS. As a result, a query on the uncertainty of data on the database level is not possible because our uncertainty vectors are encoded in binary representation. The same is, however, also true for the actual spectral data and a data selection in the SPECCHIO database is only possible via metadata [19]. The problem of querying matrices within database systems could be solved by adopting an Array Database [43], such as SciDB [44]. This would, however, not solve the handling of uncertainty data on a DBMS level. Hence, we identify a gap in available DBMSs that can handle scientific data and their associated uncertainties in matrix format.

The presented case study was chosen as a practical example of uncertainty analysis and propagation pertaining to a large number of field spectroscopy data within our in-house SPECCHIO database. These data were collected with a defined protocol over more than ten years of APEX airborne imaging spectrometer operation to allow vicarious CAL/VAL [14], [45]. The present conclusion, based on the analysis of the contribution per considered source of uncertainty, is that the main source of uncertainty is the angular alignment of the panel (see Fig. 8). This may change in the future as more sources of uncertainty get added. Nevertheless, the selected example results in an overall HCRF uncertainty of about 1.8% at a coverage factor $k = 1$. As a consequence, field spectroscopy personnel should take care to align the reference panels, achieved by the use of tripods and bubble levels [14], or possibly by gimbal-based stabilizations.

These findings once again point to the importance of field measurement protocols and to the application of rigorous uncertainty analysis to the data obtained by such protocols. Uncertainty tree diagrams and, accordingly, implementations of uncertainty propagations will only apply to specific measurement protocols and cannot easily be generalized.

A practical implication of the early standardization of our measurement protocol will be that the uncertainty propagation process, presented in this work, can be applied in retrospect to all our in situ CAL/VAL data pertaining to APEX and AVIRIS-NG [46] airborne imaging spectroscopy acquisitions. This process will be facilitated through the rich and consistent metadata set of our CAL/VAL data [19], highlighting the advantages of database-centric processing supported by SPECCHIO.

V. CONCLUSION

This work describes a substantial update of the SPECCHIO spectral information system which allows the storage of uncertainty information within the system. The newly added capabilities will support the database-centric computation of uncertainty, paving the way toward the automated computation of uncertainty for measurement data acquired following standardized protocols.

We invite the research community to make use of these new features and report any further requirements or suggested

streamlining of the SPECCHIO uncertainty API. The source code is available via GitHub [47] while the binaries can be downloaded from the SPECCHIO webpage [48] or directly from the SPECCHIO build server [49].

REFERENCES

- [1] JGCM, "GUM 2008 JCGM 100:2008 Guide to the expression of Uncertainty in Measurement," BIPM, Sevres, Paris, 2008.
- [2] H. Kostkowski, *Reliable Spectroradiometry*. La Plata, MA, USA: Spectroradiometry Consulting, 1997.
- [3] E. J. Milton, M. E. Schaepman, K. Anderson, M. Kneubühler, and N. Fox, "Progress in field spectroscopy," *Remote Sens. Environ.*, vol. 113, pp. 92–109, 2009.
- [4] L. Floridi, "In defence of the veridical nature of semantic information," *Eur. J. Anal. Philos.*, vol. 3, no. 1, pp. 31–41, 2007.
- [5] E. Woolliams, A. Hueni, and J. Gorrone, "Intermediate uncertainty analysis for earth observation (instrument calibration)," *Eur. Metrol. Res. Programme*, 2015. [Online]. Available: <http://www.emceoc.org/documents/uaoe-int-trg-course-v2.pdf>
- [6] D. R. Thompson, V. Natraj, R. O. Green, M. C. Helmlinger, B.-C. Gao, and M. L. Eastwood, "Optimal estimation for imaging spectrometer atmospheric correction," *Remote Sens. Environ.*, vol. 216, pp. 355–373, Oct. 2018, doi: [10.1016/j.rse.2018.07.003](https://doi.org/10.1016/j.rse.2018.07.003).
- [7] M. Bachmann et al., "EUFAR FP7 hyquapro (JRA-2) report on quality layers for VITO, DLR, INTO and PML," EUFAR, Toulouse, France, 2011. [Online]. Available: <http://www.zora.uzh.ch/77317/>
- [8] M. Bachmann et al., "Generic quality indicators and quality layers," EUFAR FP7 JRA2 - HYQUAPRO, 2010.
- [9] M. Bachmann, A. Makarau, K. Segl, and R. Richter, "Estimating the influence of spectral and radiometric calibration uncertainties on EnMAP data products—Examples for ground reflectance retrieval and vegetation indices," *Remote Sens.*, vol. 7, no. 8, pp. 10689–10714, 2015.
- [10] N. Carmon et al., "Uncertainty quantification for a global imaging spectroscopy surface composition investigation," *Remote Sens. Environ.*, vol. 251, Dec. 2020, Art. no. 112038, doi: [10.1016/j.rse.2020.112038](https://doi.org/10.1016/j.rse.2020.112038).
- [11] J. Pacheco-Labrador et al., "Sun-induced chlorophyll fluorescence I: Instrumental considerations for proximal spectroradiometers," *Remote Sens.*, vol. 11, no. 8, 2019, Art. no. 960, doi: [10.3390/rs11080960](https://doi.org/10.3390/rs11080960).
- [12] L. Ma et al., "Uncertainty analysis for RadCalNet instrumented test sites using the Baotou sites BTCN and BSCN as examples," *Remote Sens.*, vol. 12, no. 11, 2020, Art. no. 1696, doi: [10.3390/rs12111696](https://doi.org/10.3390/rs12111696).
- [13] J. S. Zapla-Myers, K. J. Thome, and N. P. Leisso, "Radiometric calibration of earth-observing sensors using an automated test site at Railroad Valley, Nevada," *Can. J. Remote Sens.*, vol. 36, no. 5, pp. 474–487, Oct. 2010, doi: [10.5589/m10-076](https://doi.org/10.5589/m10-076).
- [14] A. Hueni, A. Damm, M. Kneubuehler, D. Schläpfer, and M. E. Schaepman, "Field and airborne spectroscopy cross-validation—Some considerations," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 3, pp. 1117–1135, Mar. 2017, doi: [10.1109/JSTARS.2016.2593984](https://doi.org/10.1109/JSTARS.2016.2593984).
- [15] L. Sierro, "SENSECO instrument and uncertainty questionnaire results," 2020. Accessed: Nov. 17, 2020. [Online]. Available: https://www.researchgate.net/publication/344252215_SENSECO_Instrument_and_Uncertainty_Questionnaire_Results
- [16] E. Woolliams et al., "Stakeholder needs review report European metrology network for climate and ocean observation—Executive Summary," EURAMET, Braunschweig, Germany, 2021. [Online]. Available: https://www.euramet.org/secure/sdl-eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpYXQiOiJlE2NjY4NzgxNzAsImV4cCI6MTY2Njk2ODE3MCwidXNlciI6MCwiZ3Jv dXBzLjpbMCwtMV0sImZpbGUiOiJlZWZWRpYVwvZG9jc1wvRU1Oc1wvQ2xpWF0ZUz9jZWZwXuc9j29fZG9jdWllbnRzXC9TTlJlRkXhY3V0aXZlX1NlbnV1cnRFRU1OX0NPT19WZXZjZjZjZW9uXzIwMjI1MjI1e79ydl6GWd-y3C5wIpb6qmy5jq5x6r4/SNR_Executive_Summary_EMN_COO_Version_2021_01.pdf
- [17] P. De Vis and S. E. Hunt, "CoMet toolkit." Accessed: Jan. 16, 2023. [Online]. Available: <https://www.comet-toolkit.org>
- [18] J. Nightingale et al., "Ten priority science gaps in assessing climate data record quality," *Remote Sens.*, vol. 11, no. 8, 2019, Art. no. 986, doi: [10.3390/rs11080986](https://doi.org/10.3390/rs11080986).
- [19] A. Hueni et al., "The SPECCHIO spectral information system," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 5789–5799, 2020, doi: [10.1109/JSTARS.2020.3025117](https://doi.org/10.1109/JSTARS.2020.3025117).
- [20] S. Bojinski, M. Schaepman, D. Schläpfer, and K. Itten, "SPECCHIO: A Web-accessible database for the administration and storage of heterogeneous spectral data," *Photogramm. Remote Sens.*, vol. 57, pp. 204–211, 2002.
- [21] S. Bojinski, M. Schaepman, D. Schläpfer, and K. Itten, "SPECCHIO: A spectrum database for remote sensing applications," *Comput. Geosci.*, vol. 29, pp. 27–38, 2003.
- [22] A. Hueni, J. Nieke, J. Schopfer, M. Kneubühler, and K. Itten, "The spectral database SPECCHIO for improved long term usability and data sharing," *Comput. Geosci.*, vol. 35, no. 3, pp. 557–565, 2009, doi: [10.1016/j.cageo.2008.03.015](https://doi.org/10.1016/j.cageo.2008.03.015).
- [23] A. Hueni, T. Malthus, M. Kneubuehler, and M. Schaepman, "Data exchange between distributed spectral databases," *Comput. Geosci.*, vol. 37, pp. 861–873, 2011, doi: [10.1016/j.cageo.2010.12.009](https://doi.org/10.1016/j.cageo.2010.12.009).
- [24] A. Hueni, L. Suarez, L. Chisholm, and A. Held, "The use of spectral databases for remote sensing of agricultural crops," in *Hyperspectral Remote Sensing of Vegetation: Fundamentals, Sensor Systems, Spectral Libraries, and Data Mining for Vegetation*, vol. 1, P. S. Thenkabail, G. J. Lyon, and A. Huete Eds., 2nd ed. Boca Raton, FL, USA: CRC Press, 2018, p. 449.
- [25] A. Motro, "Management of uncertainty in database systems," in *Modern Database Systems: The Object Model, Interoperability, and Beyond*. Reading, MA, USA: Addison-Wesley, 1995, pp. 457–476.
- [26] R. Cheng and S. Prabhakar, "Managing uncertainty in sensor database," *Special Int. Group Manage. Data Rec.*, vol. 32, pp. 41–46, Jan. 2003, doi: [10.1145/959060.959068](https://doi.org/10.1145/959060.959068).
- [27] C. C. Aggarwal and P. S. Yu, "A survey of uncertain data algorithms and applications," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 5, pp. 609–623, May 2009, doi: [10.1109/TKDE.2008.190](https://doi.org/10.1109/TKDE.2008.190).
- [28] N. Othman, A. S. Eldin, and D. Elzanfaly, "Handling uncertainty in database: An introduction and brief survey," *Comput. Inf. Sci.*, vol. 8, no. 3, p. 119–133, Aug. 2015, doi: [10.5539/cis.v8n3p119](https://doi.org/10.5539/cis.v8n3p119).
- [29] G. Schaepman-Strub, M. Schaepman, T. H. Painter, S. Dangel, and J. V. Martonchik, "Reflectance quantities in optical remote sensing—definitions and case studies," *Remote Sens. Environ.*, vol. 103, pp. 27–42, 2006, doi: [10.1016/j.rse.2006.03.002](https://doi.org/10.1016/j.rse.2006.03.002).
- [30] J. Mittaz, C. J. Merchant, and E. R. Woolliams, "Applying principles of metrology to historical Earth observations from satellites," *Metrologia*, vol. 56, no. 3, 2019, Art. no. 032002.
- [31] P. Gege et al., "Calibration facility for airborne imaging spectrometers," *ISPRS J. Photogramm. Remote Sens.*, vol. 64, pp. 387–397, 2009.
- [32] National Institute of Standards and Technology, "NIST atomic spectra database lines form." Accessed: Mar. 3, 2022. [Online]. Available: https://physics.nist.gov/PhysRefData/ASD/lines_form.html
- [33] S. A. Trim, K. Mason, and A. Hueni, "Spectroradiometer spectral calibration, ISRF shapes, and related uncertainties," *Appl. Opt.*, vol. 60, no. 18, pp. 5405–5417, Jun. 2021, doi: [10.1364/AO.425676](https://doi.org/10.1364/AO.425676).
- [34] T. Schwarzmaier, A. Baumgartner, P. Gege, C. Köhler, and K. Lenhard, "DLR's new traceable radiance standard 'RASTA,'" in *Proc. Int. Geosci. Remote Sens. Symp.*, 2012, pp. 1–4.
- [35] D. R. Taubert et al., "Providing radiometric traceability for the calibration home base of DLR by PTB," *Amer. Inst. Phys. Conf. Proc.*, vol. 1531, no. 1, pp. 376–379, 2013, doi: [10.1063/1.4804785](https://doi.org/10.1063/1.4804785).
- [36] D. Schläpfer, J. W. Kaiser, J. Brazile, M. E. Schaepman, and K. Itten, "Calibration concept for potential optical aberrations of the APEX pushbroom imaging spectrometer," in *Proc. Int. Symp. Remote Sens. Sensors, Syst., Next Gener. Satell. IX*, 2003, vol. 5234, pp. 221–231.
- [37] H. Arndt, M. Bundschuh, and A. Naegel, "Towards a next-generation matrix library for Java," in *Proc. 33rd Annu. IEEE Int. Comput. Softw. Appl. Conf.*, 2009, vol. 1, pp. 460–467, doi: [10.1109/compsac.2009.67](https://doi.org/10.1109/compsac.2009.67).
- [38] A. Hueni and A. Bialek, "Cause, effect and correction of field spectroradiometer inter-channel radiometric steps," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 10, no. 4, pp. 1542–1551, Apr. 2017, doi: [10.1109/JSTARS.2016.2625043](https://doi.org/10.1109/JSTARS.2016.2625043).
- [39] K. Elmer, R. J. Soffer, J. P. Arroyo-Mora, and M. Kalacska, "ASDToolkit: A novel MATLAB processing toolbox for ASD field spectroscopy data," *Data*, vol. 5, no. 4, 2020, Art. no. 96, doi: [10.3390/data5040096](https://doi.org/10.3390/data5040096).
- [40] "U.K. hydrographic office and United States naval observatory," *The Nautical Almanac*, 2020 ed., 2019.
- [41] *American Practical Navigator*. Springfield, VA, USA: Nat. Geospatial-Intell. Agency, 2019.
- [42] D. Kükenbrink et al., "Mapping the irradiance field of a single tree: Quantifying vegetation-induced adjacency effects," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 7, pp. 4994–5011, Jul. 2019, doi: [10.1109/TGRS.2019.2895211](https://doi.org/10.1109/TGRS.2019.2895211).
- [43] P. Liping and D. Yanlei, "Supporting data uncertainty in array databases," in *Proc. Assoc. Comput. Machinery Special Int. Group Manage. Data Int. Conf. Manage. Data*, 2015, pp. 545–560, doi: [10.1145/2723372.2723738](https://doi.org/10.1145/2723372.2723738).

- [44] P. Brown, "Overview of SciDB: Large scale array storage, processing and analysis," in *Proc. Assoc. Comput. Machinery Special Int. Group Manage. Data Int. Conf. Manage. Data*, 2010, pp. 963–968.
- [45] M. Schaepman et al., "Advanced radiometry measurements and Earth science applications with the airborne prism experiment (APEX)," *Remote Sens. Environ.*, vol. 158, pp. 207–219, 2015.
- [46] W. J. Chapman et al., "Spectral and radiometric calibration of the next generation airborne visible infrared spectrometer (AVIRIS-NG)," *Remote Sens.*, vol. 11, no. 18, 2019, Art. no. 2129, doi: [10.3390/rs11182129](https://doi.org/10.3390/rs11182129).
- [47] SPECCHIO, "SPECCHIO source code," Accessed: Dec. 7, 2022. [Online]. Available: <https://github.com/SPECCHIODB/SPECCHIO>
- [48] SPECCHIO, "SPECCHIO spectral information system," Accessed: Dec. 7, 2022. [Online]. Available: www.specchio.ch
- [49] SPECCHIO, "SPECCHIO jenkins server," Accessed: Dec. 7, 2022. [Online]. Available: <https://jenkins.specchio.ch>



Andreas Hueni (Member, IEEE) received the B.Sc. degree in computer science from the University of Applied Science Brugg-Windisch, Windisch, Switzerland, in 1997, the P.G.Dipl. in geographic information systems and the M.Phil.(Sc.) degree in earth science from Massey University, Palmerston North, New Zealand, in 2005 and 2006, respectively, and the Ph.D. degree in geography from the University of Zurich (UZH), Zurich, Switzerland, in 2011.

He has worked on the APEX project since 2007 and is currently an Academic Associate with the Remote Sensing Laboratories, UZH, responsible for APEX sensor and data calibration, and the Project Manager with the Airborne Research Facility for the Earth System. His research interests include the calibration of spectrometers, uncertainty analysis, and the design of combined database and software systems, such as the spectral database SPECCHIO.



Kimberley Mason received the M.Sc. degree in geophysics from Imperial College London, London, U.K., in 2016.

Since 2019, she has been working with the Remote Sensing Laboratories, University of Zurich, Zurich, Switzerland, as a Scientific Programmer. Her current projects include SPECCHIO, the spectral information system, as well as image processing, sensor calibration, and uncertainty analysis.



Simon Trim received the B.Sc. degree in physical sciences in 2012 and the M.Sc. degree in space sciences in 2015 from the University of Liège, Liège, Belgium. He is currently working toward the Ph.D. degree with the Remote Sensing Laboratories, University of Zurich, Zurich, Switzerland.

He is currently a Research Scientist with ReSe Applications LLC, Wil, Switzerland. His research interests include the overarching issue of hyperspectral data traceability and related uncertainties, spectroradiometric calibration and validation of spectrometers, and related applications for earth observation.